REMARKS

Claims 1-15 are pending in this application. All have been rejected under 35 U.S.C. 102(e) in view of the published application to Muret. Applicant asks the Office to reconsider this application and allow all of the claims.

Muret does not show, nor does he even suggest, a system in which data is loaded from the transaction logs of Internet servers "across plural parallel processing modules" of a database system. Likewise, Muret does not show nor suggest executing "a database query across [such] parallel processing modules" to select all of the data associated with a particular user session.

Muret does indeed describe a database system that sifts through Internet web-log data in an attempt to reconstruct individual user sessions, but Muret's system is quite different than that described and claimed by Applicant. In particular, Muret's system uses a very complex, and inevitably slow, sequential program to parse through web-log data, distributing this data across a vast array of tables. These tables include a "visitor table 310" and many "data tables 315," which themselves include a variety of nested tables, such as a "hash table 340," a "rank table 345," a "record table 350," and a "string table 355." (Muret, ¶¶ [0060] - [0061].) The complexity and sequential nature of the control program is clear from Muret's description of the modules that it includes (e.g., a "buffer update module 240" and a "log parser module 210") and the data-management pieces that it must oversee (e.g., "pre-allocated log buffer 600" and a "pointer array 610"). (See Muret, ¶¶ [0072], [0080] – [0081].) Muret himself describes the control program as one having loops nested within loops: "The control routine comprises a main loop 1200, a visitor loop 1210 nested within the main loop 1200, and a read loop 1215 nested within the visitor loop 1210." (¶ [0057].)

This system of Muret's is nothing like Applicant's, which leverages the power of plural parallel processing modules to simplify and dramatically speed up the process of identifying individual user sessions within web-log data. The complexity of Muret's multiple program modules and data tables is replaced in Applicant's system by simple database-query commands. For example, much of the function performed by the "log

parser module" in Muret's system (¶ [0078] *et seq.*) can be carried out in a system like
Applicant's with the following simple Structured Query Language (SQL) code:

```
INSERT INTO presession
SELECT (SUBSTR(weblog_txt,21,(INDEX(weblog_txt,'{')-21)))
, (SUBSTR(weblog_txt,2,9)(DATE, FORMAT 'MM/DD/YY')(INTEGER)) +
            (SUBSTR(weblog_txt,11,8)(FLOAT, FORMAT '99:99:99')(INTEGER'
      ,(SUBSTR(weblog_txt,(INDEX(weblog_txt,'{')),300))
FROM inputtest.   (Applicant's disclosure, page 5.)
```

Likewise, all of the buffering, hashing, and pointing that is done in Muret's
system is unnecessary in a system like Applicant's, where similar function is carried out
through the execution of simple database-query commands across the parallel processing
modules of the database system.

Simply put, Muret does not show, nor does he even suggest, executing "a
database query across plural processing modules," as claimed by Applicant.
Accordingly, all of Applicant's claims are patentable over the Muret reference.

CONCLUSION

All of Applicant's claims are allowable over the art of record. Applicant asks the Office to reconsider this application and allow all of the claims. Please apply any charges that might be due, excepting the issue fee but including fees for extensions of time, to deposit account 50-1673.

Respectfully,

John D. Cowart
Reg. No. 38, 415

NCR Corporation
1700 South Patterson Blvd.
Dayton, Ohio 45479-0001

Tel. No. (858) 485-4903
Fax No. (858) 485-2581